AN2146
APPLICATION NOTE

STR71xF Embedded Flash Programming with ADS

INTRODUCTION

The purpose of this application note is to provide **ARM Developer Suite (ADS)** users with information to help them program binary files to Flash memory using the software's Flash downloader utility. This document provides step-by-step instructions for converting an application in ELF format to simple binary format and then downloading the resulting binary to the internal Flash memory of the STR71xF microcontroller on a target application board.

Necessary software and hardware tools

- **ARM Developer Suite** version 1.2, a complete software development solution for creating applications for ARM core-based microcontrollers that includes the AXD debugger.
- **Flash downloader** (flash.li) for STR71xF, the modified version of the default Flash downloader, included with this application note, which allows the identification and programming of STR71xF devices.

Note: The default integrator version of the Flash downloader that is used to program other standard CFI-type Flash devices is furnished in binary format in the directory: ***install_directory\ADS\bin\flash.li****.*

You may want to keep a copy of the default version of flash.li in another directory or with another name if you work with devices other than the STR71xF.

- **STR710 Evaluation Board** (STR710-EVAL) or other evaluation/application board featuring an STR71xF microcontroller.
- **ARM MULTI-ICE server** and interface unit (JTAG standard interface)

Note: It is also possible to use ARM Angel Debug Monitor (running from RAM) and a serial or parallel connection to the host PC. For more information about Angel, refer to the ARM Software Development Toolkit User Guide.

1 Overview of Flash programming with ADS

When you invoke the Flash downloader utility (flash.li) from the AXD debugger, the debugger downloads this utility into RAM on the target board. The microcontroller then executes the Flash programming routine, retrieving the binary file from the host PC and then programming it to Flash memory.

The Flash downloader utility (flash.li) provided with this application note has been adapted by ST Microelectronics to allow programming of STR71xF microcontrollers. This modified version must be used when programming your STR71xF device, instead of the default integrator version of the Flash downloader that is provided with ADS.

2 Programming procedure

The following section takes you step by step through the process of downloading a binary file to Flash memory.

The STR710-Eval board is used as an example to illustrate the hardware set up, however the same fundamental procedure can be applied to other application/evaluation boards featuring ST ARM core-based microcontrollers.

The example in this procedure also refers to ARM Multi-ICE, however it is possible to use the ARM Angel Debug Monitor. For more information refer to the *ARM Software Development Toolkit User Guide*.

2.1 Converting ELF image file into plain binary format

The file that you will program to internal Flash must be in plain binary format suitable for direct download into Flash memory. You can convert your image files in Executable Linkable Format (ELF) to plain binary format using the fromELF utility.

To convert an ELF file to a binary (.bin) file, invoke the fromELF utility using the command line as shown below:

```
fromelf -bin -o outfile.bin infile.axf
```

***-bin**: specifies the output file format*

***-o outfile.bin**: specifies the name of the output file*

***infile.axf**: specifies the ELF file to be translated*

For a detailed explanation of the fromELF command line and options, you can refer to the *ARM Developer Suite Linker and Utilities Guide*.

2.2 Configuring and connecting to hardware

The target board should be configured to boot from RAM prior to invoking the Flash downloader utility. This is to ensure that the memory where the Flash programming routine will be downloaded is mapped to RAM.

If you are using the STR710-Eval board, configure the boot mode as follows:

1 Configure switches **SW13**, **SW14** and **SW15** to RAM Mode (RAM mapped at 0h).

Table 1: STR710-EVAL switch configuration

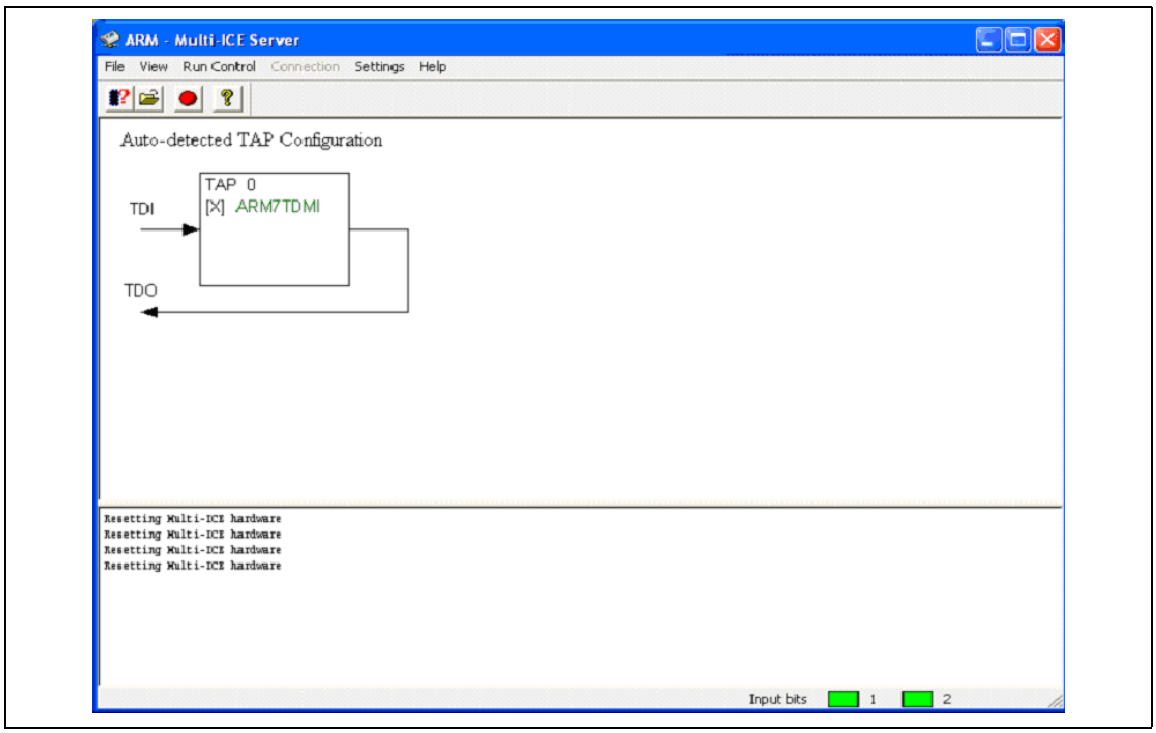
Switch	Description		Position
• **SW13**	Boot mode 1:	Execute code from internal RAM RAM mapped at 0h	**2-3**
• **SW14**	Boot mode 0:		**1-2**
• **SW15**	Boot mode EN:		**2-3**

Note: You can refer to STR710-Eval Board Datasheet for detailed descriptions of switch positions and boot modes.

2 Connect the target board to your host PC via the Multi-ICE interface device and power on your target board.

3 Start the Multi-ICE server and select ***File>Auto Configure***. Multi-ICE server connects to and identifies the target. Once the target is identified, the window will be similar to that shown in *Figure 1*.

Figure 1: Multi-ICE server connected to target board



Note: If Multi-ICE server cannot identify the target you will have to do a manual configuration. Refer to the ARM Multi-ICE User Guide.

2.3 Using the Flash downloader from AXD

Once you have connected to the target you can launch AXD and run the Flash Downloader utility. From the AXD interface, you can now invoke the Flash downloader utility and download your binary file to the target.

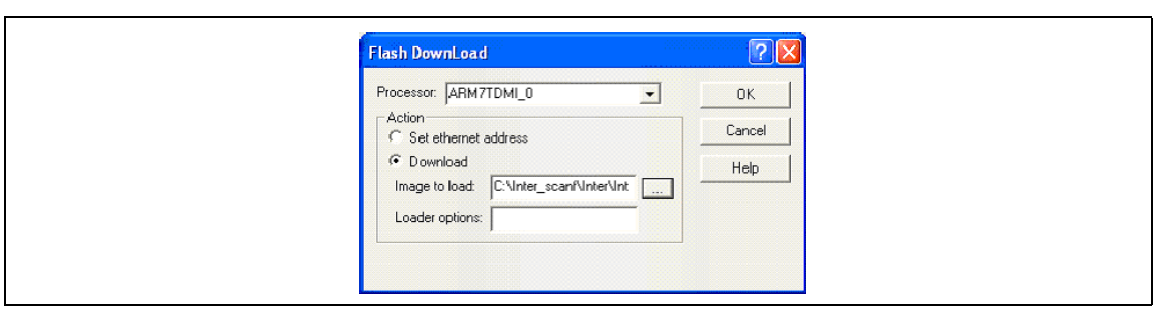
To do so:

1 Copy the furnished **flash.li** file for STR71xF devices to the directory *Install_directory\ADS\bin*

2 Start AXD.

3 In AXD select ***File>Flash Download...*** The **Flash DownLoad** dialog box is displayed (see *Figure 2*)

Figure 2: Flash DownLoad dialog box



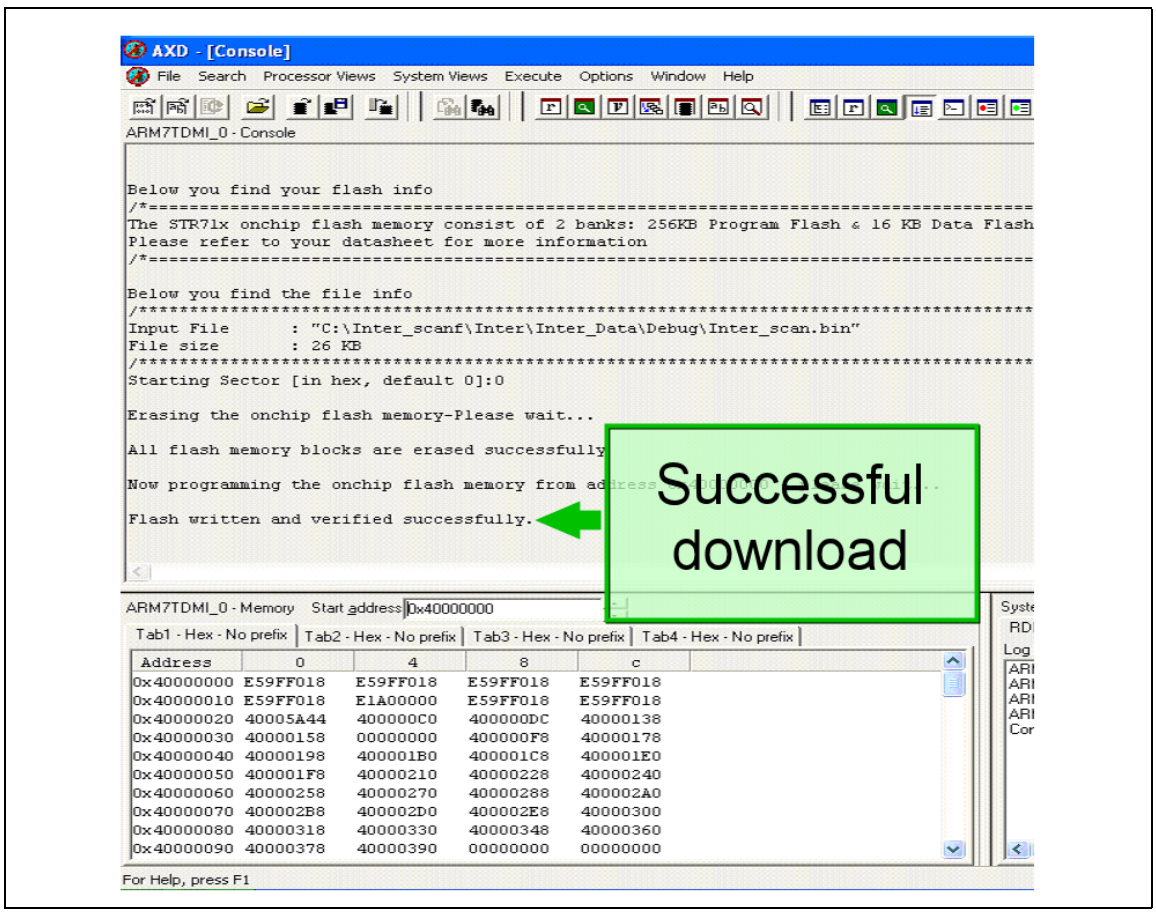
4 Specify the pathname for your binary file, or click ***Browse*** to select the file.

Note: If the path name contains spaces you must enclose the pathname in quotes.

5 Click ***OK***. The Flash downloader reads your binary file and displays the download settings in the Console processor view.

6 You can edit the settings if required, or press ***Enter***. AXD will display a message when the data has been written to flash (see *Figure 3*).

Figure 3: Successful download to Flash memory


AXD - [Console]
File Search Processor Views System Views Execute Options Window Help
ARM7TDMI_0 - Console
Below you find your flash info
The STR71x onchip flash memory consist of 2 banks: 256KB Program Flash & 16 KB Data Flash
Please refer to your datasheet for more information
Below you find the file info
Input File : "C:\Inter_scanf\Inter\Inter_Data\Debug\Inter_scan.bin"
File size : 26 KB
Starting Sector [in hex, default 0]:0
Erasing the onchip flash memory-Please wait...
All flash memory blocks are erased successfully
Now programming the onchip flash memory from ad
Flash written and verified successfully.
Successful download
ARM7TDMI_0 - Memory Start address 0x40000000
Tab1 - Hex - No prefix Tab2 - Hex - No prefix Tab3 - Hex - No prefix Tab4 - Hex - No prefix
Address 0 4 8 c
0x40000000 E59FF018 E59FF018 E59FF018 E59FF018
0x40000010 E59FF018 E1A00000 E59FF018 E59FF018
0x40000020 40005A44 400000C0 400000DC 40000138
0x40000030 40000158 00000000 400000F8 40000178
0x40000040 40000198 400001B0 400001C8 400001E0
0x40000050 400001F8 40000210 40000228 40000240
0x40000060 40000258 40000270 40000288 400002A0
0x40000070 400002B8 400002D0 400002E8 40000300
0x40000080 40000318 40000330 40000348 40000360
0x40000090 40000378 40000390 00000000 00000000
For Help, press F1

Note: The Flash downloader lets you choose the start sector to write in. If you choose a start sector other than zero, your image will be loaded in the specified sector, but when you boot from the on chip Flash memory you will not see your program running.

7 If you downloaded your program at starting sector 0, then change the switch positions to boot from Flash memory (see *Table 2*) and reboot, you can see your program running.

Table 2: STR710-EVAL switch configuration

Switch	Description		Position
• **SW13**	Boot mode 1:	System executes a “primary boot loader (ST-firmware)” from Boot-FLASH then jumps to RAM BOOTFLASH mapped at 0h	**1-2**
• **SW14**	Boot mode 0:		**2-3**
• **SW15**	Boot mode EN:		**2-3**

3 For further information...

All the latest news, documentation and software downloads for STR7 microcontrollers and development tools can be found at **www.st.com/mcu**.

If you require further information about this procedure or the ARM tools or utilities referred to in this document, you can download the following supporting documents at **www.arm.com**:

- ARM Application Note 111 Flash Programming
- ARM Developer Suite Linker and Utilities Guide
- ARM Multi-ICE User Guide
- ARM Software Development Toolkit User Guide

4 Revision History

Table 3: AN2146 Revision History

Date	Revision	Description
01-May-2005	1.0	• Initial release

The present note which is for guidance only, aims at providing customers with information regarding their products in order for them to save time. As a result, STMicroelectronics shall not be held liable for any direct, indirect or consequential damages with respect to any claims arising from the content of such a note and/or the use made by customers of the information contained herein in connection with their products.

Information furnished is believed to be accurate and reliable. However, STMicroelectronics assumes no responsibility for the consequences of use of such information nor for any infringement of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of STMicroelectronics. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. STMicroelectronics products are not authorized for use as critical components in life support devices or systems without express written approval of STMicroelectronics.

The ST logo is a registered trademark of STMicroelectronics.

All other names are the property of their respective owners

© 2005 STMicroelectronics - All rights reserved

STMicroelectronics GROUP OF COMPANIES

Australia - Belgium - Brazil - Canada - China - Czech Republic - Finland - France - Germany - Hong Kong - India - Israel - Italy - Japan - Malaysia - Malta - Morocco - Singapore - Spain - Sweden - Switzerland - United Kingdom - United States of America

www.st.com